Intruder Detection (updated for combat deployment)

by Anthony of Boston

Wait for the model to load before clicking the button to enable the webcam - at which point it will become visible to use. The app will beep when an intruder is detected. The app will also emit voice alert, saying "intruder detected" upon detection of an intruder. This app allows phones to be mounted in various places, to which it can detect when an intruder is in the area. This can be used in clearing and counter insurgency operations. This is also useful for the civilian population against thugs and other criminal elements in urban environments. The app can prevent ambush attacks. The phone could be placed in the cracks of walls and other discreet locations. With internet access, a user could view the scene remotely with facebook live and see when intruders have gained unauthorized access.

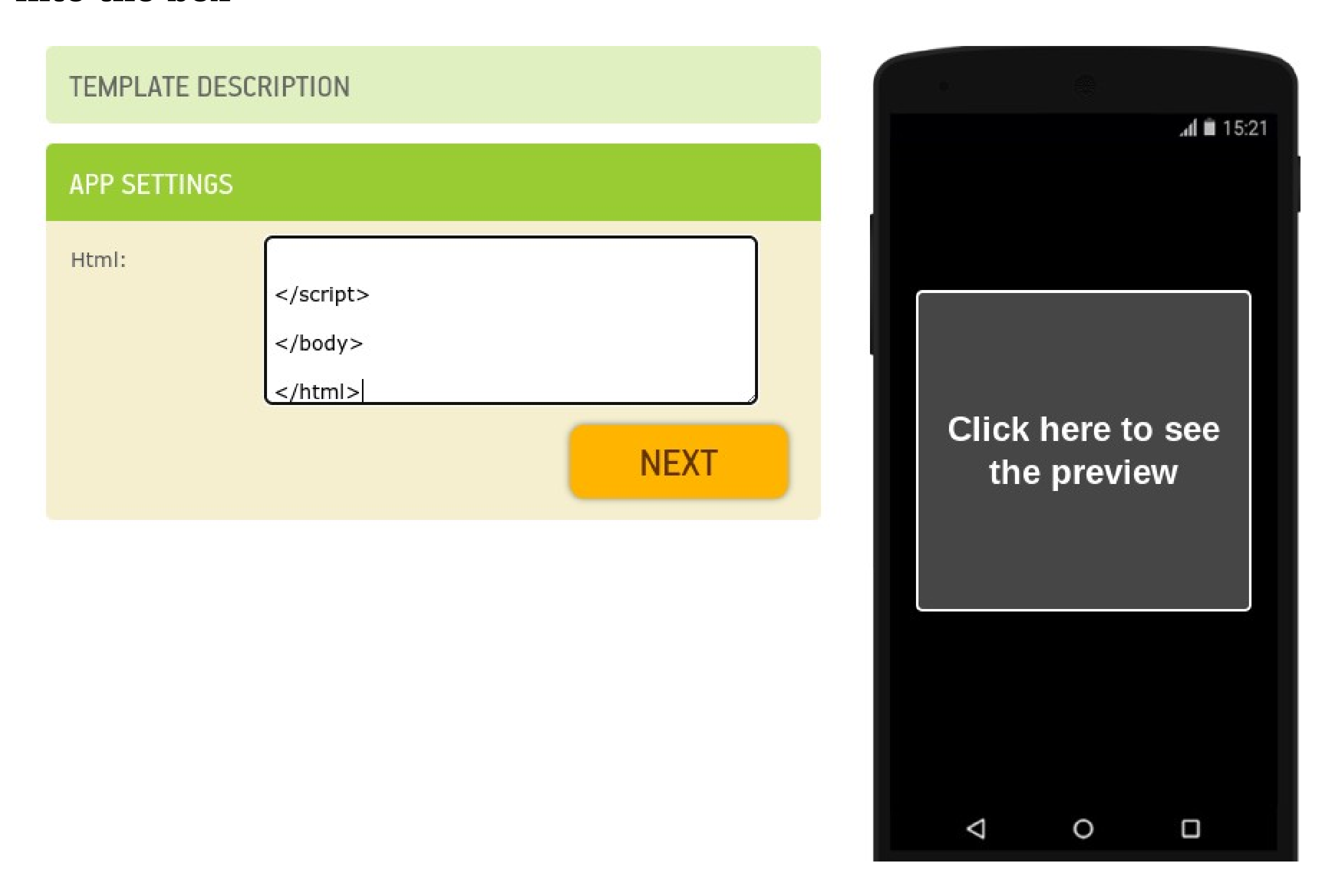




From page 7 to page 15, is the HTML code for the Intruder Detection App that you can copy and paste and test in html:

In order to create a cell phone app, first download this doc file and then copy and paste the html code that starts on page 7. Then go to https://appsgeyser.com/create-html-app/

When you get to the website, click next and then paste the HTML code into the box

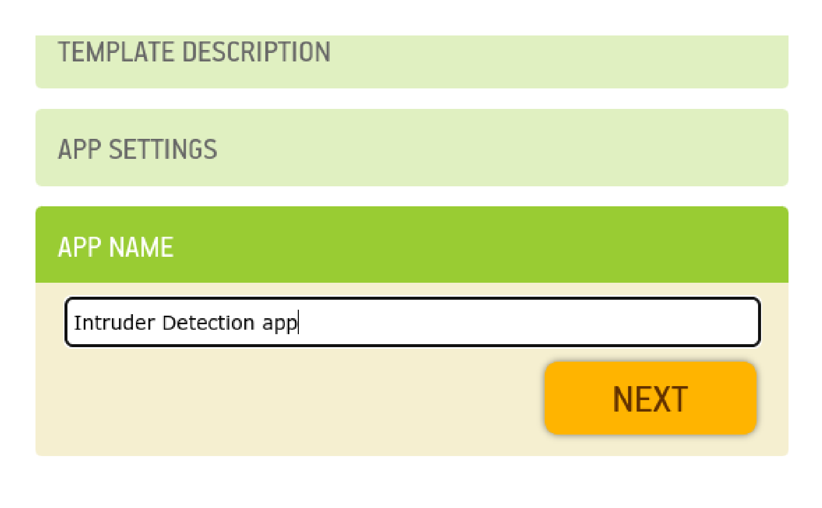


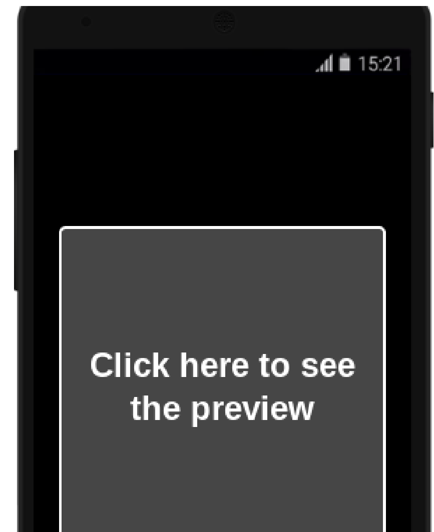
You can preview the app on the right by clicking the gray box which says "Click here to see preview."

Otherwise, click "Next"

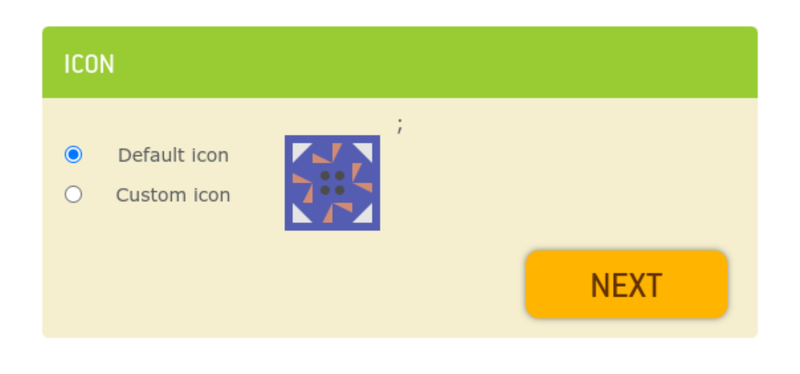
Then add a title to your "Intruder" app

During this process, you will be asked to open an account. Register or use gmail to sign up.



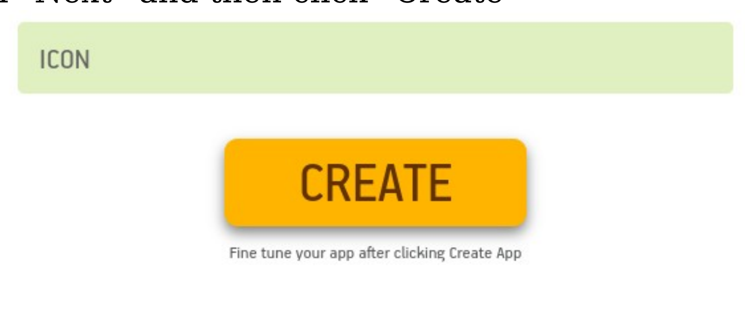


Click “Next”, and then choose Icon



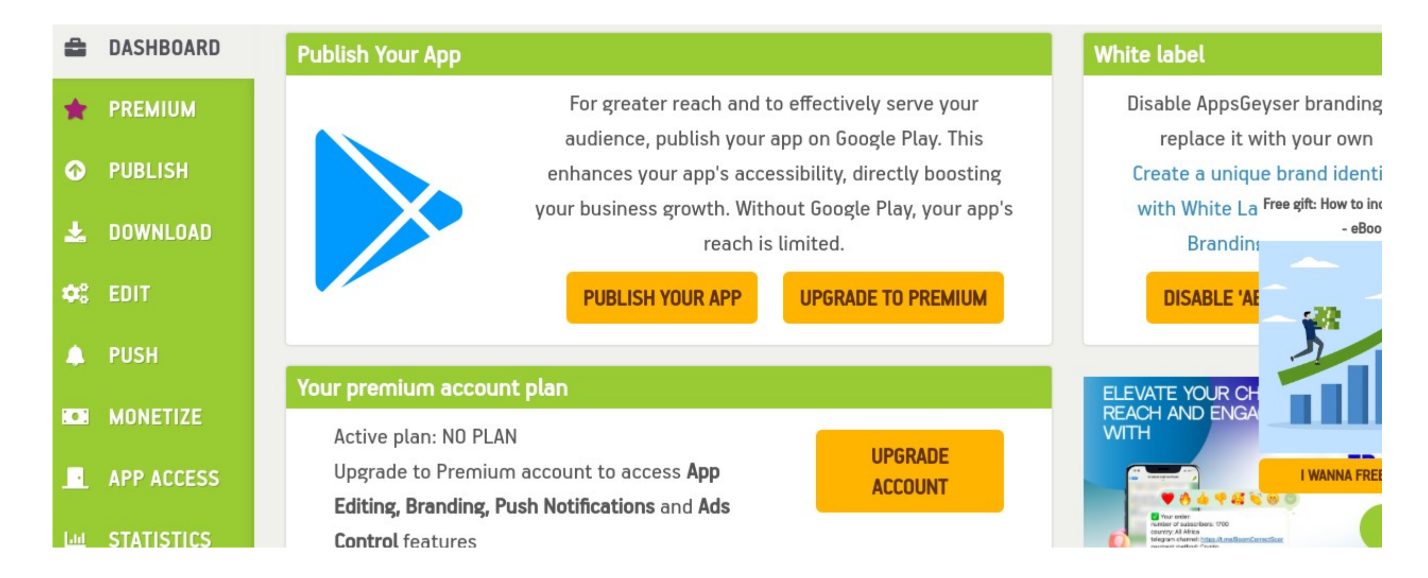


Click “Next” and then click “Create”

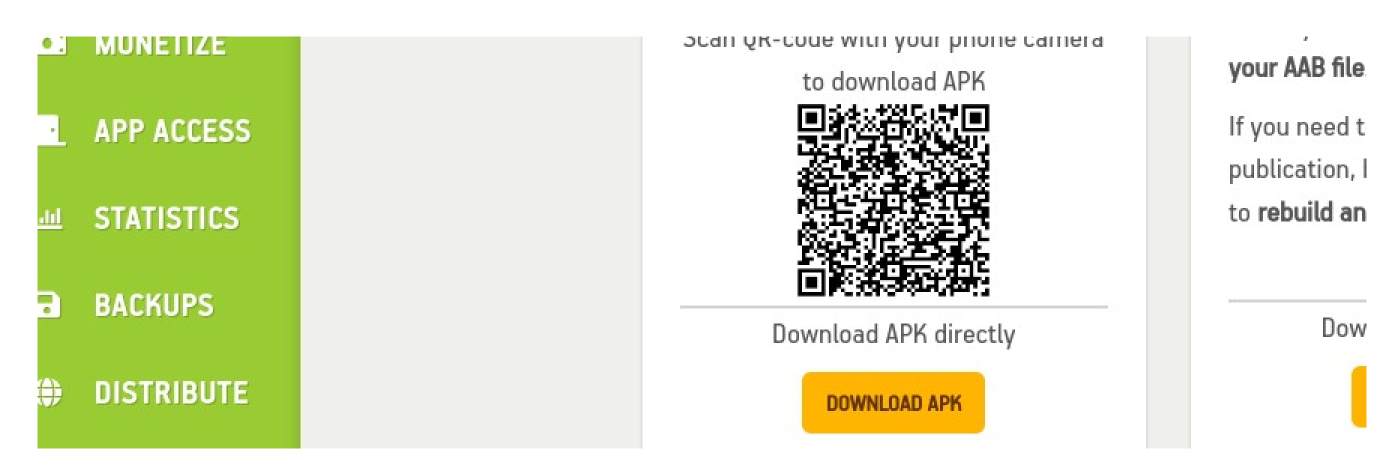




Now click “Download” on the left side of the screen



On the next page, click “Download APK”



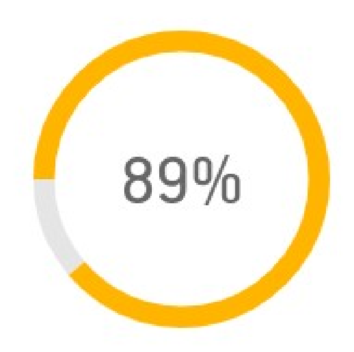
Let the app finish building and follow instructions from there.



Your app is building!

Please, wait 2-5 minutes...

Signing app



Please Note that the app may not work on “Samsung” devices, but should perform on Android devices.

To continue installation, the user has allow the phone to install unknown apps. This can be done by going into settings and making the change.

Afterwards, a google play alert may pop up indicating that the app is unsafe and has been blocked.
Simply bypass this by clicking the “install anyway” option.

Once installed, the app should be ready for deployment. I cannot guarantee that this will work on any phone, however, and should seen as a beta version.

On the next page is the HTML code.

```
<!DOCTYPE html>
<html>
<head>
</head>
<body style="background-color: transparent;">

</body>
</html><html lang="en">

<head>

<title>Intruder Detection</title>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="stylesheet" href="style.css">

</head>

<style>



body {


width: 100%;

height:100%;

font-family: 'Open Sans', sans-serif;



color: #000;

font-size: 10px;

text-align: center;

letter-spacing: 1.2px;
```

```
}

h1 {

font-style: bold;

color: #ffffff;

}

video {

    position: fixed; right: 0; bottom: 0;
    min-width: 100%; min-height: 100%;
    width: auto; height: auto; z-index: -100;

    background-size: cover;

}

section {

opacity: 1;

transition: opacity 500ms ease-in-out;

}

.removed {

display: none;

}

.invisible {

opacity: 0.2;

}

.camView {
```

```
position: relative;
float: center;
text-align: center;
width: calc(100% - 20px);
margin: 10px;
cursor: pointer;
}

.camView p {
position: absolute;
padding: 5px;
background-color: rgba(255, 0, 0, 0.85);
color: #FFF;
border: 1px rgba(255, 0, 0, 0.7);
z-index: 2;
font-size: 17px;
}

.highlighter {

border: 4px dashed #ff0000;
z-index: 1;
position: absolute;
}</style>

<body>

<p id="del"> <b>Intruder Detection</b><br /><br />

Wait for the model to load before clicking the button to enable the webcam - at
which point it will become visible to use. The app will beep when an intruder is
detected. The app will also emit voice alert, saying "intruder detected" upon
detection of an intruder. This app allows phones to be mounted in various places,
to which it can detect when an intruder is in the area. This can be used in
clearing and counter insurgency operations. This is also useful for the civilian
population against thugs and other criminal elements in urban environments.
The phone could be placed in the cracks of walls and other discreet locations.
With internet access, a user could view the scene remotely with facebook live
and see when intruders have gained unauthorized access.</p>


<p style = visibility: hidden id="textID"> Intruder Detected</p>


<div id="countdown"></div>


<section id="demos" class="invisible">
```

```
<p id="demo"></p>

<div id="liveView" class="camView">

<button id="webcamButton" onclick="start1()"  >Enable Webcam</button>

<video id="webcam" autoplay width="640" height="480"></video>

</div>

</section>



</div>

<p><br> </p>

<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs/dist/tf.min.js"
type="text/javascript"></script>

<script
src="https://cdn.jsdelivr.net/npm/@tensorflow-models/coco-ssd"></script>

<script src="script.js" defer></script>
<script> function start1() {document.getElementById("del").innerHTML = "";
var e = document.getElementsByTagName('html')[0];
}</script>

<script>const video = document.getElementById('webcam');

const liveView = document.getElementById('liveView');

const demosSection = document.getElementById('demos');

const enableWebcamButton = document.getElementById('webcamButton');

var model = undefined;

var children = [];

function getUserMediaSupported() {

return !!(navigator.mediaDevices &&

navigator.mediaDevices.getUserMedia);

}
```

```
if (getUserMediaSupported()) {

enableWebcamButton.addEventListener('click', enableCam);

} else {

console.warn('getUserMedia() is not supported by your browser');

}

function enableCam(event) {

if (!model) {

return;

}

event.target.classList.add('removed');

const constraints = {

video: true

};

navigator.mediaDevices.getUserMedia(constraints).then(function(stream) {

video.srcObject = stream;

video.addEventListener('loadeddata', predictWebcam);

});

}


var beep = (function () {

var ctxClass = window.audioContext ||window.AudioContext ||
window.AudioContext || window.webkitAudioContext

var ctx = new ctxClass();

return function (duration, type, finishedCallback) {


duration = +duration;
```

```
type = (type % 5) || 0;

if (typeof finishedCallback != "function") {
finishedCallback = function () {};
}

var osc = ctx.createOscillator();

osc.type = type;

osc.connect(ctx.destination);
if (osc.noteOn) osc.noteOn(0);
if (osc.start) osc.start();

setTimeout(function () {
if (osc.noteOff) osc.noteOff(0);
if (osc.stop) osc.stop();
finishedCallback();
}, duration);

};
})();

function textToSpeech() {
const speech = new SpeechSynthesisUtterance();
let voices = speechSynthesis.getVoices();
let convert = document.getElementById("textID").innerHTML;
```

```
speech.text = convert;

speech.volume = 1;
speech.rate = 0.9;
speech.pitch = 0;

speech.voice = voices[0];

speechSynthesis.speak(speech);
}


function pause() {
window.speechSynthesis.pause();
}


function stop() {
window.speechSynthesis.cancel();
}


cocoSsd.load().then(function (loadedModel) {
model = loadedModel;
demosSection.classList.remove('invisible');
});

function predictWebcam() {

model.detect(video).then(function (predictions) {
```

```
for (let i = 0; i < children.length; i++) {
liveView.removeChild(children[i]);
}
children.splice(0);
window.speechSynthesis.pause();
for (let n = 0; n < predictions.length; n++) {

if ( predictions[n].class == "person") {
predictions[n].class = "Intruder Detected"
window.speechSynthesis.resume();
textToSpeech();
beep(1000, 2, function () {
});


const p = document.createElement('p');
p.innerText = predictions[n].class + ' - with '
+ Math.round(parseFloat(predictions[n].score) * 100)
+ '% confidence.';
p.style = 'margin-left: ' + predictions[n].bbox[0] + 'px; margin-top: '
+ (predictions[n].bbox[1] - 10) + 'px; width: '
+ (predictions[n].bbox[2] - 10) + 'px; top: 0; left: 0;';

const highlighter = document.createElement('div');
highlighter.setAttribute('class', 'highlighter');
```

```
highlighter.style = 'left: ' + predictions[n].bbox[0] + 'px; top: '
+ predictions[n].bbox[1] + 'px; width: '
+ predictions[n].bbox[2] + 'px; height: '
+ predictions[n].bbox[3] + 'px;';

liveView.appendChild(highlighter);
liveView.appendChild(p);
children.push(highlighter);
children.push(p);

}
else{
}
}
window.requestAnimationFrame(predictWebcam);
});
}


</script>
</body>
</html>
```